
Documentação do retorno XML referente projeto Busca Endereço por CEP

Neste documento você encontrará a estrutura do XML que o web service retorna e alguns exemplos de como usar.

Códigos de Retorno para controle das informações:

- 001 - CLIENTE INATIVO OU INEXISTENTE!**
- 002 - CEP ENCONTRADO COM SUCESSO!**
- 003 - CEP NÃO ENCONTRADO!**
- 004 - CLIENTE NÃO POSSUÍ BILHETAGEM CADASTRADA PARA ESTE MÊS/ANO!**
- 005 - LIMITE DE ACESSO BILHETAGEM ATINGIDO PARA ESTE MÊS/ANO!**

001 – Se o código do Cliente não existir ou Cliente estiver inativo.

002 – Quando o CEP informado existir.

003 – Quando o CEP informado não existir.

004 – Quando o Cliente não tiver um controle cadastrado para um determinado Serviço “**Busca Endereço por CEP**” num período de mês/ano. Neste caso a Interfix deverá ser contatada para atualizar o cadastro.

005 – Quando o Cliente atingir o limite máximo de consulta de Serviço “**Busca Endereço por CEP**”, e o seu contrato indicar que neste momento seu acesso às consultas estarão indisponíveis para este mês, sendo liberado novamente no próximo mês ou no mesmo, caso haja liberação por parte da Interfix após uma renegociação.

OBS: Ao receber o XML contendo as informações necessárias, ler primeiramente os campos:

```
<IDRESULTADO></IDRESULTADO >  
<RESULTADO></RESULTADO>
```

Pois são estes campos que recebem os códigos acima.

Ex:

```
<IDRESULTADO>001</IDRESULTADO>  
<RESULTADO>CLIENTE INATIVO OU INEXISTENTE!</RESULTADO>
```

Importante

Se o código retornado for diferente de “002” – “**CEP ENCONTRADO COM SUCESSO!**”, a leitura do restante do XML poderá ser desconsiderada, visto que as informações que contemplarão este XML não são utilizáveis nem relevantes, ou seja apenas quando o código for **002** o XML deverá ser lido por completo para recuperar as informações do Endereço retornado na busca por CEP.

À seguir estão todos os XML's gerados, lembrando que o único que será lido por completo será o que retornar no campo **<IDRESULTADO>** o valor **002**.

URL do Webservice: <http://www.autocep.com.br/webcep/wsEndereco.asmx?WSDL>

Nome da Função: [AutenticaClienteXml](#)

Parâmetros da Função:

- idCliente: (int)
- chave: (varchar(15))
- CEP: (varchar(8))

1 - Estrutura XML do retorno

O retorno do web service pode retornar 5 resultados diferentes, mas sempre na mesma estrutura de XML, veja os tipos de retorno:

- 001 - CLIENTE INATIVO OU INEXISTENTE!
- 002 - CEP ENCONTRADO COM SUCESSO!
- 003 - CEP NÃO ENCONTRADO!
- 004 - CLIENTE NÃO POSSUI BILHETAGEM CADASTRADA PARA ESTE MÊS/ANO!
- 005 - LIMITE DE ACESSO BILHETAGEM ATINGIDO PARA ESTE MÊS/ANO!

Obs: A estrutura xml será sempre a igual, mesmo que o retorno da busca por CEP não for encontrada, veja nas estruturas do XML que o web service retorna abaixo:

```
<?xml version="1.0" encoding="utf-8" ?>
<ListaEnderecoCEP>
  <EnderecoCEP>
    <IDRESULTADO>001</IDRESULTADO>
    <RESULTADO>CLIENTE INATIVO OU INESISTENTE!</RESULTADO>
    <CEP></CEP>
    <TIPO> </TIPO>
    <CHAVE></CHAVE>
    <UF></UF>
    <TITULO> </TITULO>
    <PREPOSICAO/><PREPOSICAO>
    <NOME> </NOME>
    <BAIRRO1> </BAIRRO1>
    <BAIRRO2></BAIRRO2>
    <CEPMIN></CEPMIN>
    <CIDADE> </CIDADE>
    <COD_DDD></COD_DDD>
    <COD_CIDADE></COD_CIDADE>
  </EnderecoCEP>
</ListaEnderecoCEP>
```

```
<?xml version="1.0" encoding="utf-8" ?>
<ListaEnderecoCEP>
  <EnderecoCEP>
    <IDRESULTADO>002</IDRESULTADO>
    <RESULTADO>CEP ENCONTRADO COM SUCESSO!</RESULTADO>
    <CEP>1006040</CEP>
    <TIPO>PC</TIPO>
    <CHAVE>01000</CHAVE>
```

```
<UF>SP</UF>
<TITULO />
<PREPOSICAO />
<NOME>PAULO ALFEU DE MONTEIRO DUARTE</NOME>
<BAIRRO1>SE</BAIRRO1>
<BAIRRO2 />
<CEPMIN>01000000</CEPMIN>
<CIDADE>SAO PAULO</CIDADE>
<COD_DDD>011</COD_DDD>
<COD_CIDADE>9668</COD_CIDADE>
</EnderecoCEP>
</ListaEnderecoCEP>
```

```
<?xml version="1.0" encoding="utf-8" ?>
<ListaEnderecoCEP>
  <EnderecoCEP>
    <IDRESULTADO>003</IDRESULTADO>
    <RESULTADO>CEP NÃO ENCONTRADO!</RESULTADO>
    <CEP></CEP>
    <TIPO> </TIPO>
    <CHAVE></CHAVE>
    <UF></UF>
    <TITULO> </TITULO>
    <PREPOSICAO/><PREPOSICAO>
    <NOME> </NOME>
    <BAIRRO1> </BAIRRO1>
    <BAIRRO2></BAIRRO2>
    <CEPMIN></CEPMIN>
    <CIDADE> </CIDADE>
    <COD_DDD></COD_DDD>
    <COD_CIDADE></COD_CIDADE>
  </EnderecoCEP>
</ListaEnderecoCEP>
```

```
<?xml version="1.0" encoding="utf-8" ?>
<ListaEnderecoCEP>
  <EnderecoCEP>
    <IDRESULTADO>004</IDRESULTADO>
    <RESULTADO>CLIENTE NÃO POSSUI BILHETAGEM CADASTRADA
    PARA ESTE MÊS/ANO!</RESULTADO>
    <CEP></CEP>
    <TIPO> </TIPO>
    <CHAVE></CHAVE>
    <UF></UF>
    <TITULO> </TITULO>
    <PREPOSICAO/><PREPOSICAO>
    <NOME> </NOME>
    <BAIRRO1> </BAIRRO1>
```

```
<BAIRRO2></BAIRRO2>
<CEPMIN></CEPMIN>
<CIDADE> </CIDADE>
<COD_DDD></COD_DDD>
<COD_CIDADE></COD_CIDADE>
</EnderecoCEP>
</ListaEnderecoCEP>
```

```
<?xml version="1.0" encoding="utf-8" ?>
<ListaEnderecoCEP>
  <EnderecoCEP>
    <IDRESULTADO>005</IDRESULTADO>
    <RESULTADO>LIMITE DE ACESSO BILHETAGEM ATINGIDO PARA
    ESTE MÊS/ANO!</RESULTADO>
    <CEP></CEP>
    <TIPO> </TIPO>
    <CHAVE></CHAVE>
    <UF></UF>
    <TITULO> </TITULO>
    <PREPOSICAO/><PREPOSICAO>
    <NOME> </NOME>
    <BAIRRO1> </BAIRRO1>
    <BAIRRO2></BAIRRO2>
    <CEPMIN></CEPMIN>
    <CIDADE> </CIDADE>
    <COD_DDD></COD_DDD>
    <COD_CIDADE></COD_CIDADE>
  </EnderecoCEP>
</ListaEnderecoCEP>
```

2 – Exemplo (Asp.Net)

Neste exemplo você verá o web service sendo consumido através de Asp.net.

Crie as seguintes funções abaixo:

```
Imports System.Globalization
Imports System.Data
Imports System.Data.SqlClient
Imports System.Xml
Imports System.Text
Imports System.IO

Private Function FnStrXmlToDataset(ByVal strArqXml As String)
    Dim xmlSR As System.IO.StringReader = New
System.IO.StringReader(strArqXml)
    Dim ds As New Data.DataSet
    ds.ReadXml(xmlSR)
    Return ds
End Function

Public Function LerXML(ByVal strXML As String) As String
    Dim strRet As String = "" ' Proc que procura no xml por
determinado objetos , passando pra respectivas variaveis
    Dim ds As Data.DataSet
    ds = FnStrXmlToDataset(strXML)
    Dim sNome As String = ""
    Dim intTables As Integer = 0
    Dim intRows As Integer = 0
    Dim intCols As Integer = 0

    For intTables = 0 To ds.Tables.Count - 1
        strRet += " table = " & ds.Tables(intTables).TableName & " "
        Table = ds.Tables(intTables).TableName
        For intRows = 0 To ds.Tables(intTables).Rows.Count - 1
            For intCols = 0 To ds.Tables(intTables).Columns.Count - 1
                strRet += " intCols(" & intCols & ") = " &
ds.Tables(intTables).Rows(intRows).Item(intCols) & " " & "<br>"
            Next
        Next
    Next
    Return strRet
End Function
```

Esta função abaixo é a que irá interpretar o web service para distribuir os valores na tela.

```
Public Sub BuscaCep(ByVal CEP As String)

    'tira o traço do CEP para realizar a busca
    Dim pCEP As String
    pCEP = Replace(CEP, "-", "")

    'instancia o dataset
    dsRetorno = New DataSet

    'invoca o metodo AutenticaCliente
```

```
dsRetorno = WsEndereco.AutenticaCliente(txtCodigoCliente.Text,  
txtChave.Text, CEP)
```

```
Dim retXML As String  
retXML = dsRetorno.GetXml
```

```
LerXML(retXML)
```

```
Dim flag As Integer  
flag = 0
```

```
If Table = "CEP" Then
```

```
    If dsRetorno.Tables("CEP").Rows.Count > 0 Then  
        For Each dr As DataRow In dsRetorno.Tables("CEP").Rows
```

```
            If dsRetorno.Tables("CEP").Rows.Count = 1 Then
```

```
                txtCodigoRetorno.Text = dr("IDRESULTADO")  
                txtDescricaoRetorno.Text = dr("RESULTADO")
```

```
            ElseIf dsRetorno.Tables("CEP").Rows.Count = 2 Then
```

```
                If flag = 0 Then
```

```
                    flag = 1
```

```
                    If IsDBNull(dr("TIPO")) Then  
                        txtTipo.Text = ""
```

```
                    Else  
                        txtTipo.Text = dr("TIPO")
```

```
                    End If  
                    If IsDBNull(dr("TITULO")) Then  
                        txtTitulo.Text = ""
```

```
                    Else  
                        txtTitulo.Text = dr("TITULO")
```

```
                    End If  
                    If IsDBNull(dr("PREPOSICAO")) Then  
                        txtPreposicao.Text = ""
```

```
                    Else  
                        txtPreposicao.Text = dr("PREPOSICAO")
```

```
                    End If  
                    If IsDBNull(dr("NOME")) Then  
                        txtNome.Text = ""
```

```
                    Else  
                        txtNome.Text = dr("NOME")
```

```
                    End If  
                    If IsDBNull(dr("BAIRRO1")) Then  
                        txtBairro1.Text = ""
```

```
                    Else  
                        txtBairro1.Text = dr("BAIRRO1")
```

```
                    End If  
                    If IsDBNull(dr("BAIRRO2")) Then  
                        txtBairro2.Text = ""
```

```
                    Else  
                        txtBairro2.Text = dr("BAIRRO2")
```

```
                    End If  
                    If IsDBNull(dr("CEPMIN")) Then
```

```
        txtCep.Text = ""
    Else
        txtCep.Text = dr("CEPMIN")
    End If
    If IsDBNull(dr("CIDADE")) Then
        txtCidade.Text = ""
    Else
        txtCidade.Text = dr("CIDADE")
    End If
    If IsDBNull(dr("UF2")) Then
        txtUF.Text = ""
    Else
        txtUF.Text = dr("UF2")
    End If
    If IsDBNull(dr("COD_DDD")) Then
        txtDDD.Text = ""
    Else
        txtDDD.Text = dr("COD_DDD")
    End If
    If IsDBNull(dr("COD_CIDADE")) Then
        txtCodigoIBGE.Text = ""
    Else
        txtCodigoIBGE.Text = dr("COD_CIDADE")
    End If

    Else
        txtCodigoRetorno.Text = dr("IDRESULTADO")
        txtDescricaoRetorno.Text = dr("RESULTADO")
    End If

End If

Next

End If

Else

Dim flag2 As Integer
flag2 = 0
If dsRetorno.Tables("CLI").Rows.Count > 0 Then
    For Each dr As DataRow In dsRetorno.Tables("CLI").Rows

        If dsRetorno.Tables("CLI").Rows.Count = 2 Then
            If flag2 = 1 Then

                txtCodigoRetorno.Text = dr("IDRESULTADO")
                txtDescricaoRetorno.Text = dr("RESULTADO")
                Exit For

            End If
            flag2 = 1
        Else
            txtCodigoRetorno.Text = dr("IDRESULTADO")
            txtDescricaoRetorno.Text = dr("RESULTADO")
            Exit For
        End If

    Next

End If
```

```
End If
```

```
End Sub
```

3 – Exemplo (PHP)

Neste exemplo você verá o web service sendo consumido através do PHP.

Crie as seguintes funções ou visualize no exemplo abaixo:



Ws Php.rar

```
<?php
```

```
    // Parametros de consulta (id e chave são fornecidos pela Interfix), o
    CEP é o que se deseja consultar.
    $idCliente = $_REQUEST["pIdCliente"];
    $chave = $_REQUEST["pChave"];
    $cep = $_REQUEST["pCep"];

    // carrega os parametros que serão passados para o web service
    $parametros = array();
    $parametros['idCliente'] = $idCliente;
    $parametros['chave'] = $chave;
    $parametros['CEP'] = $cep;

    try
    {
        // URL do web service
        $url="http://www.autocep.com.br/webcep/wsEndereco.asmx?WSDL";

        // Instancia o web service
        $WS = new SoapClient($url, $parametros);

        // Invoca o Metodo do web service
        $result = $WS->AutenticaClienteXml($parametros);

        $xml = simplexml_load_string($result->AutenticaClienteXmlResult->any);

        echo "<pre>" ;

        foreach ($xml as $key => $value)
        {
            foreach($value as $ekey => $eValue)
            {
                print($ekey . " = " . $eValue . PHP_EOL);
            }
        }

    } catch ( SoapFault $fault ) {
```

```
trigger_error ( "SOAP Fault: (faultcode: {$fault->faultcode},  
faultstring: {$fault->faultstring})", E_USER_ERROR );  
}
```

?>